Appendix I

Extended Squitter Enhanced Reception Techniques

This page intentionally left blank.

I Extended Squitter Enhanced Reception Techniques

I.1 Purpose and Scope

The purpose of this Appendix is to provide a description of improved squitter reception techniques. Elements of improved squitter reception include (1) the use of amplitude to improve bit and confidence declaration accuracy, (2) more capable error detection/correction algorithms, (3) more selective preamble detection approaches, and (4) combinations of the above.

The improved techniques presented in this Appendix represent one way of achieving the performance requirements specified in §2.2.4.4 for enhanced squitter reception. The squitter processing configuration used as the basis for these performance requirements is specified in Section I.5.

The reception techniques, as required in §2.2.4.3.4 of this MOPS, are also described in this appendix for comparison with the enhancements.

I.2 Background

Squitter reception includes the detection of the Mode S 1090 MHz waveform preamble, declaration of the bit and confidence values, error detection, and (if necessary) error correction. The current techniques for squitter reception are based upon techniques developed for use in Mode S narrow-beam interrogators and for TCAS. In both of these applications, the rate of Mode A/C fruit (see glossary) that is stronger than the Mode S waveform is relatively low, nominally less than 4,000 fruit per second.

Early applications investigated for Extended Squitter, prior to the development of this document, included long range air-ground surveillance, surface surveillance and support for TCAS. Of these three applications, the only one with the potential for operating in significantly higher fruit environments was the air-ground application. For this application, it is possible to use sectorized antennas (6 to 12 sectors) to limit the amount of fruit detected by any receiver.

Extended Squitter applications have now been defined in DO-242A, including long range (up to 90 NM) air-air surveillance in support of free flight. This type of surveillance is referred to as Cockpit Display of Traffic Information (CDTI). For this application, sectorized antennas are not an option. In high-density environments, it is possible to operate with fruit rates of 40,000 fruit per second and higher.

Even with these fruit rates, Extended Squitter reception using current techniques will provide useful performance. However, operation of Extended Squitter in very high Mode A/C fruit environments has led to the development of improved squitter reception techniques to support long range CDTI in high-density environments.

I.3 Current Squitter Reception Techniques

Receiving systems for Mode S replies and squitters are currently implemented operationally in ground based Mode S interrogator/receiver systems and in TCAS

avionics. For TCAS, the current reception techniques are defined in DO-185A. These current reception techniques are the basis for the reception requirements in this MOPS (section 2.2). The enhanced techniques that are the subject of this Appendix are not required in this version of the MOPS, but are planned to be required in a future version.

I.3.1 Overview of Current Techniques

Extended squitter uses the 112-bit Mode S waveform shown in Figure I-1. Other than bit assignments, this is the same as the long reply waveform currently used operationally for air-ground replies and for air-air coordination messages in TCAS. This waveform encodes data using pulse position modulation (PPM). A chip in the leading half of the bit position represents a binary ONE. A chip in the trailing half bit position represents a binary ZERO. Reception of the Squitter begins with the detection of the four-pulse preamble. At preamble detection a dynamic threshold is set 6 dB below the level of the preamble. Any signal received below this threshold will not be seen by the squitter reception processor. This eliminates the effect of low level Mode A/C and Mode S fruit on the reception process.

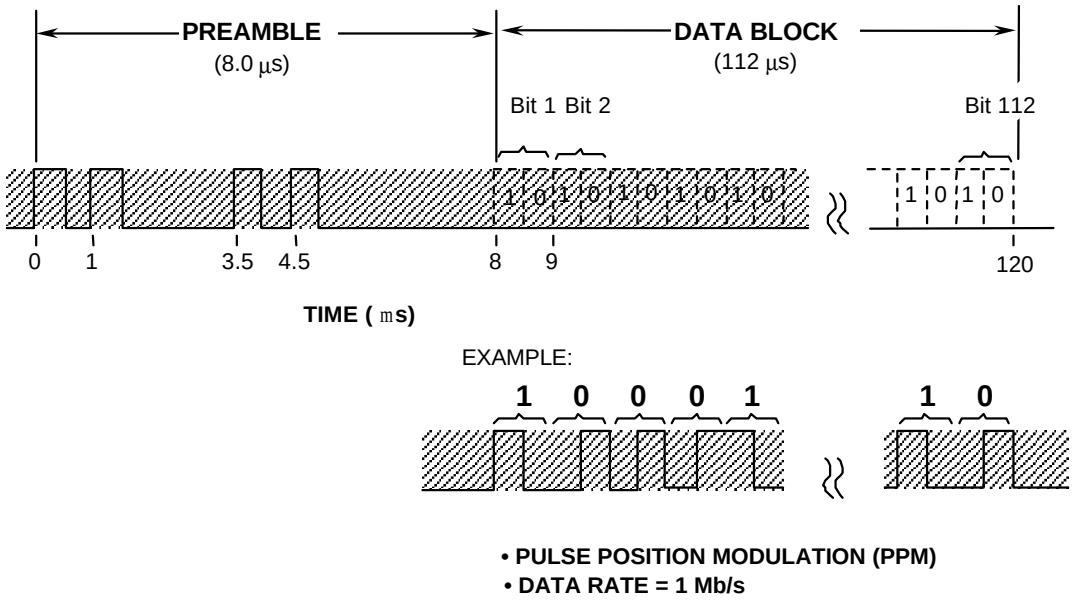

Figure I-1: Mode S Extended Squitter Waveform

Once all of the bits have been received, error detection is performed using the 24-bit CRC contained in the PI field. If no error is detected, the squitter is passed on to surveillance processing. If an error is detected (indicated by a non-zero error syndrome) an error correction technique is applied. The current error correction algorithm can correct the errors caused by one stronger overlapping Mode A/C fruit.

I.3.2 Current Techniques for Bit and Confidence Declaration

Currently, bit values are declared by comparing the amplitudes of the centers of the two chips; the chip with the greater amplitude is declared the bit value. This amplitude

comparison technique limits bit errors caused by fruit of lower level than the squitter being received (Figure I-2, parts a and b).

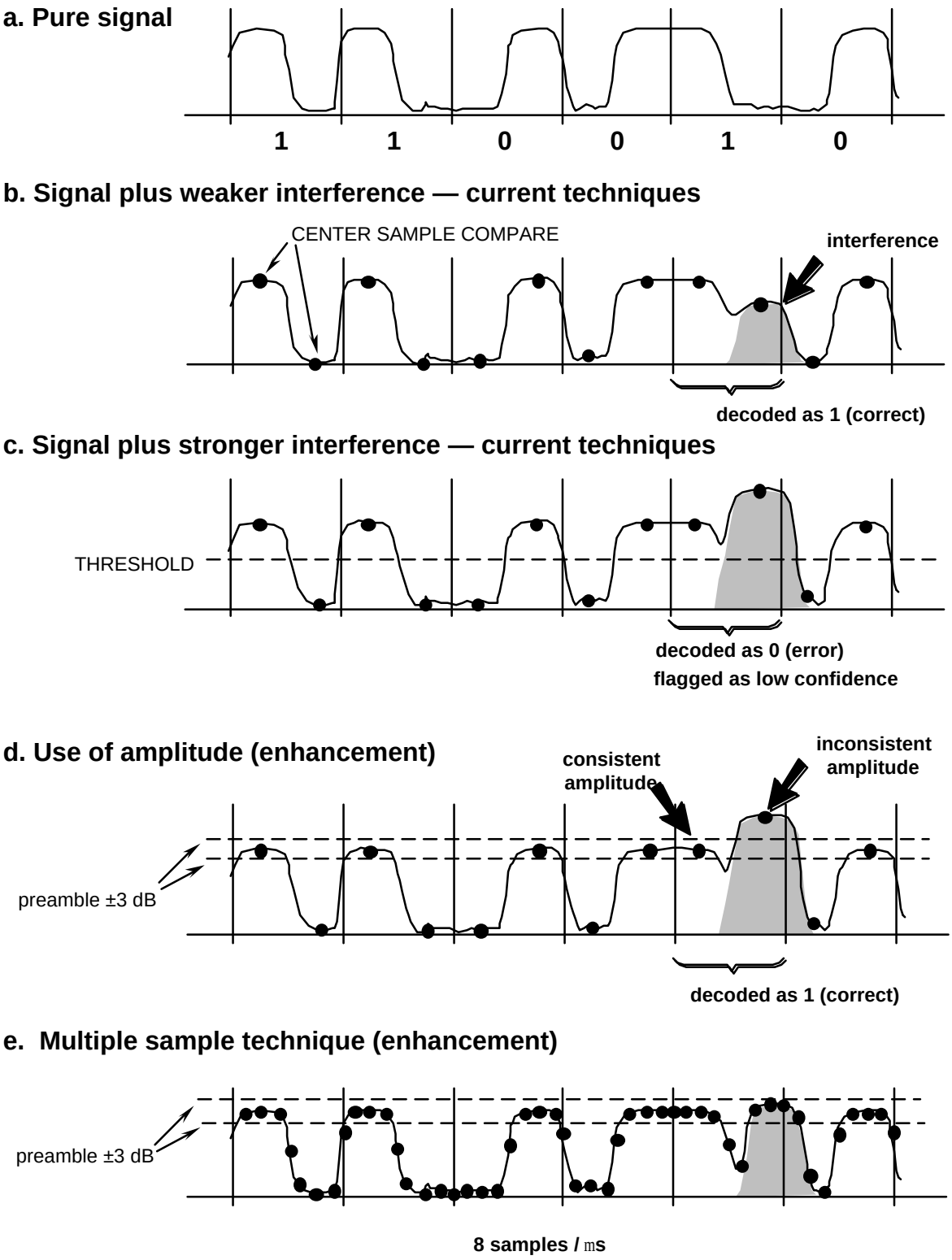

Figure I-2: Current and Enhanced Bit Demodulation Techniques

Note: *The vertical scale represents log video signals.*

Currently confidence for each bit is declared by observing if signals are above threshold on both chips. If only one chip is above threshold, it is declared to be high confidence since there is no evidence of overlapping fruit. If both chips have amplitude above threshold, low confidence is declared since some form of interference had to be present to cause this condition (Figure I-2, part c).

Since the nominal width of a Mode A/C pulse is 0.45 microsecond, an interfering Mode A/C pulse can affect only 1 of the 2 chips of a Mode S bit position. If the chip affected is the one with the Mode S pulse, the level of the pulse in the chip will be altered, but since it is the only chip with energy, the bit will be declared as high confidence. If instead the interfering pulse affects the blank chip, pulses will exist in both positions and the Mode S bit will be declared as low confidence. Since the bit decision is based on the larger pulse, if the Mode A/C fruit is stronger than the Mode S signal, any such bit will be declared in error, while if the Mode S signal is stronger, any such bit will be declared correctly.

I.3.3 Current Error Detection and Correction Techniques

I.3.3.1 Overview

When a Mode A/C fruit interferes with a Mode S Extended Squitter, some of the Mode S bits may be declared in error. The current Mode S error correction algorithm attempts to correct these errors by locating a string of 24 consecutive bits such that the low confidence bits in the window cover the bits in the corresponding error syndrome [reference I-1]. If such a window is found, all low confidence bits matching the syndrome are flipped, and the message is declared correct.

I.3.3.2 Sliding Window Technique

This algorithm operates by examining successive 24 bit windows, starting with bits 89-112 of the message. In order to achieve a successful error correction, each message bit in the window corresponding to a one (1) in the error syndrome must have its value complemented (i.e., a one is changed to a zero (0) and a zero is changed to a one). This complementing can only be done if each of the bits is declared to be low confidence. If so, each of the bits is complemented and the message is declared to be error corrected. If not, the window is shifted one bit downward, a transformed syndrome is computed and the process repeats. The process ends when a correctable error pattern has been found, or the sliding window reaches the beginning of the message. In order to control undetected errors, correction is not attempted if there are more than 12 low confidence bits in the window.

This technique provides error correction in cases where a Mode S message has been overlaid with one stronger Mode A/C fruit (that caused all bit errors) and one or more weaker Mode A/C fruit that were above the dynamic threshold (and caused only low confidence bits). This technique is well suited to the low levels of Mode A/C fruit observed in a narrow-beam Mode S interrogator or a TCAS. This technique is not

appropriate for high fruit rate environments since it produces a high undetected error rate; for this reason, its use is prohibited in §2.2.4.4.

I.4 Enhanced Squitter Reception Techniques

Throughout the following description, a sampling rate of 8 samples per microsecond is used. This is a typical value, although other, higher, sampling rates may be used.

I.4.1 Enhanced Preamble Detection

I.4.1.1 Background

When a heavy Mode A/C fruit environment exists, squitter preambles are often lost due to the existence of earlier apparent preambles produced by Mode A/C fruit pulses. Since preamble detection can be suppressed for up to 130 microseconds after each preamble declared, a Mode S squitter starting during the false preamble dead time will be lost. Also, if the time computed for the preamble is offset by even one sample count, the correction performance of the multiple amplitude sample method (described in §I.4.2.3) in heavy fruit is severely compromised.

I.4.1.2 Enhanced Preamble Declaration Techniques

Two general methods of overcoming the loss of Mode S preambles in the dead time following the declaration of apparent preambles due to the coincident timing of Mode A/C pulses are:

1. tighten the preamble declaration process to prevent Mode A/C-induced preambles from being declared, through (§I.4.1.2.1) reducing the allowed time variation in the detection of the defined four-pulse preamble and (§I.4.1.2.2.3) to require the existence of additional pulses in the preamble declaration process.

2. use multiple or re-triggerable Mode S preamble detectors in order to detect the presence of a strong preamble that occurs while processing a weaker squitter. Note that a re-triggerable detector is the simpler approach.

In particular, this approach requires the detection of some or all of the first five data pulses of the Mode S signal (the DF field) in their proper data positions in order to declare a valid preamble, thereby in effect increasing the number of preamble pulses from 4 to as many as 9. For an extended squitter reception, there must be a pulse in at least one of the two chip positions of the five bits of the DF Field. To satisfy this test, the pulses must be at least 3 samples wide.

I.4.1.2.1 Preamble Time Correction

The current preamble detection specification permits a ±1 sample period (period = 1/8 microsecond) variation in the alignment of the leading edge positions of the preamble pulses. Should the first preamble pulse have its leading edge obscured by Mode A/C interference, it is likely that the declared preamble time will be set 1 sample too early, while destructive interference can lead to a declaration time 1 sample late. Simulation

results indicate that this effect causes a significant degradation in performance in a high interference environment.

To counteract this problem, the corrective action is to forbid both +1 (late) and -1 (early) variations in the preamble declaration process. That is, if a pulse requires a -1 offset, no later pulse can use the +1 offset. Similarly if a pulse requires a +1 offset, no later pulse can use the -1 offset. Once this is accomplished, the next step is to note the positions of all clear leading edges of the preamble pulses following the first. If the majority occur in the +1 (or -1) position, the declared time of the preamble is adjusted accordingly. Simulation results indicate this fix removes virtually all of the performance degradation.

I.4.1.2.2 Preamble Declaration Process

I.4.1.2.2.1 Overview

The following description is based on a sampling rate of 8 samples per microsecond, whose phasing is asynchronous relative to incoming signals. Hence, each Mode S preamble pulse and each data chip is nominally seen by 4 data samples. Each data sample is assumed to have a digitized amplitude value ranging from the receiver noise level to the maximum signal level expected, precise to a fraction of a dB. A threshold is set for the receiver; samples below this level are not detected.

A key element of all the algorithms is the location of pulses and their leading edges. At an 8 MHz sampling rate, a pulse consists of 3 or more successive samples above threshold. A “valid pulse position” is defined to be any sample that is both above threshold and is followed by 2 other samples above threshold. If an implementation uses a sampling rate greater than 8 MHz, a pulse should be determined by the number of samples above threshold that is best suited to detecting pulses that have a minimum width of 0.3 microseconds.

At an 8 MHz sampling rate, a leading edge position is any sample within a declared pulse, that is above threshold and is 6 dB or more greater than its preceding sample and less than 6 dB lower than its succeeding sample. In general, the test value is 48 dB divided by the number of samples per microsecond. Thus a pulse can contain several successive valid pulse positions as well as 2 or more leading edges (never in succession); a minimum pulse might have only 1 valid pulse position and 0 leading edges.

The improved preamble detection algorithm includes three steps that are described in the following sections of this Appendix:

a. Preamble detection (§I.4.1.2.2.2)

b. Preamble validation (§I.4.1.2.2.3)

c. Reference level generation (§I.4.1.2.2.4)

These steps can be performed in the order a-b-c, or alternatively in the order a-c-b. Both designs have been evaluated and found to perform effectively.

The preamble detection step is used to identify, by locating a 4-pulse preamble, the potential starting point of a set of samples to be processed as an extended squitter. The preamble validation step prunes the list of preambles in an attempt to eliminate most false preamble detections caused by interference. The reference level generation step determines a ”reference level” that is an estimate of the received signal power level, that is used (like DMTL) in subsequent message bit processing and decoding. If step c is performed before step b, then the reference level can be used in the validation step.

I.4.1.2.2.2 Preamble Detection

The current Extended Squitter preamble detection algorithm sets a dynamic threshold at 6 dB below the amplitude level of the first pulse of a potential 4-pulse preamble. All subsequent pulses in the preamble (and the later data block bits) must exceed this amplitude to be declared. The location of the reference point of each of the 4 preamble pulses must be a declared valid pulse position. The sample time of the first preamble pulse reference point defines the timing for the decoding of the Mode S message. To declare a Mode S preamble, 4 pulse reference points must exist in the waveform with the appropriate preamble spacing, with at least two of the reference points being declared leading edges. A plus or minus one sample timing tolerance is allowed for each of the subsequent preamble pulse positions relative to the first pulse reference.

There are two limitations of the current Mode S preamble detection algorithm when operated in an environment with high rates of Mode A/C fruit. First, overlapping Mode A/C fruit replies can generate false preambles. Triggering on false preamble detections shuts down the preamble detection circuitry for 64 or 120 microseconds during which a valid Extended Squitter waveform may be lost. Second, an overlapping high-level Mode A/C fruit pulse could sufficiently raise the amplitude of the first Mode S preamble pulse such that the dynamic threshold is set so high that the rest of the Mode S preamble pulses are not detected. The improved algorithm is designed to mitigate these effects.

Detection of a valid 4 pulse preamble is performed similarly to the current algorithm. The exceptions are as follows:

1. The improved Extended Squitter preamble detection algorithm does not use a dynamic threshold level. All pulses above the receiver threshold are detected.

2. The improved reception technique does not use inferred leading edges in preamble detection.

3. Pulse sample timing tolerance is limited to either one sample plus or one sample minus but not both in the same preamble (as described above in §I.4.1.2.1).

4. If two or more of the subsequent pulses have leading edges that represent a timing offset (either plus 1 or minus 1) relative to the first pulse reference point, the reference time for the preamble is shifted one clock in the appropriate direction (as described above in §I.4.1.2.1). This helps to compensate for distortion of the first Mode S preamble pulse due to overlapping Mode A/C interference. It also helps to insure that the sampling of the pulses in the data block is done more accurately.

Note that the improved Mode S preamble detection algorithm can be less strict than the current algorithm in that it does not utilize a dynamic threshold. Thus the four pulses in the potential Mode S preamble do not need to be of similar amplitude in order to pass the improved algorithms tests. The next processing step is intended to minimize false Mode S preamble detections that occur due to multiple Mode A/C fruit replies.

I.4.1.2.2.3 Preamble Validation

The improved preamble detection algorithm is actually less discriminating than the current Mode S algorithm. The current preamble detection algorithm would eliminate too many real Mode S preambles if the Mode A/C fruit rate is high. However, the improved preamble detection algorithm is quite susceptible to false preamble detections caused by high interference rates. The preamble validation step reduces the rate of false preamble detections to a very low level by looking for pulses in the first five bits of the data block area of the potential Extended Squitter waveform (the DF field). The PPM encoding used in the Extended Squitter data block requires that there be a pulse in either the 0 or 1 chip of each data block position. This is a pattern that is difficult to generate with Mode A/C fruit replies.

Ideally, the algorithm could attempt to find pulses in all 112 data positions. However, Mode A/C fruit interference could make it difficult to accurately find all the pulses. Simulation and experimentation have shown that high fruit rate environments often destroy one or more data pulses in an Extended Squitter waveform. Trying to qualify all the data pulse positions would also entail quite a bit of processing time. A trade-off is to check for the pulses in the first 5 bits of the data block -- the Extended Squitter data format (DF) field. Since proper decoding of the DF field is essential to decoding the rest of the data block (i.e., the first bit of the DF field determines whether the data block has 112 bits, this check of the DF field pulses is highly likely to eliminate waveforms that would not have been decodable (either because they were actually false preamble detections or because they are too corrupted by Mode A/C fruit).

The DF field processing searches for valid pulse positions (see definition in §I.4.1.2.2.1) in either the start of the 1 chip or the start of the 0 chip. The search in either case allows a plus or minus 1 sample offset to allow for sample clock timing jitter. If all five DF data pulse positions have valid pulses in them, the potential Extended Squitter waveform is declared valid. If any of the five DF pulses are not valid, the potential Extended Squitter waveform is rejected. (Note that with sufficiently high levels of Mode A/C fruit, it is possible that a false Extended Squitter waveform can get by even the improved detection algorithm. However, with high probability, this false message will be rejected by error detection and correction.)

I.4.1.2.2.4 Reference Level Generation

The reference level generation process seeks to find the appropriate signal amplitude for the Extended Squitter waveform Extended Squitter waveform, and then to set a dynamic threshold 6 dB below the reference. The process is an extension of the current Mode S dynamic threshold operation which simply takes the amplitude of the first preamble pulse as the reference, and that level minus 6 dB as the cutoff for all subsequent pulses in the Extended Squitter waveform Extended Squitter waveform. The dynamic threshold

process is susceptible to corruption of the first preamble pulse by overlapping Mode A/C fruit, leading to an incorrect (probably too high) setting for the dynamic threshold and the potential loss of the Extended Squitter waveform Extended Squitter waveform. The reference level generation processing described here uses all the amplitude information in the preamble (all four pulses) to set the amplitude level for the Mode S message. This more accurate reference level can also be used to do a more effective job of decoding the Mode S message bits by comparing their amplitudes to the reference level.

The reference level generation process begins by selecting amplitude samples from each of the preamble pulses that are considered appropriate candidates, namely those that have leading edges declared in their reference position. Limiting the sampling to only those preamble pulses with leading edges declared in the proper positions tends to remove from consideration those preamble pulses that are most likely corrupted by overlapping Mode A/C pulses. The two amplitude samples after the leading edge for each qualified preamble pulse are entered into the reference level declaration algorithm; these are the samples most likely to be in the flat section of the pulse. If a sampling rate higher that 8 MHz is used it may be required to use 2 samples later than those immediately following the leading edge of the pulse to insure sampling in the flat section of the pulse.

The reference level generation process continues with the set of qualified preamble pulse amplitudes (up to 8 samples are possible). For each qualified sample amplitude, a search over all qualified sample amplitudes is performed to generate a count of how many amplitudes lie within plus or minus 2 dB of the given sample amplitude. If the highest count sample is unique (no ties with other samples), then the reference level is the winner's amplitude. Any ties are broken using the following procedure. If the amplitude of the samples in the tied set differs by more than 2 dB, remove all samples greater than 2 dB above the lowest sample in the tied set. (The assumption is that higher amplitudes are more likely to be the result of overlapping Mode A/C fruit corrupting the sample.) Then, take the average of all remaining samples from the tied set to be the reference level.

I.4.1.2.3 Re-triggerable Preamble Detection

It is quite possible for a strong Mode S Extended Squitter to arrive during the time that a weaker signal is being decoded. When that event occurs, it is probably desirable, for two reasons, to shift processing to the new signal: (1) stronger signals represent closer aircraft and hence greater threats; (2) the overlapping stronger signal will most likely prevent proper decoding of the original signal. In order to accomplish re-triggering, the preamble decoding circuitry must be left active during the decoding phase of processing. Then, when a stronger preamble is detected, the processing can be re-started for the new signal.

The preamble detection logic for re-triggering is the same as that specified above with one major exception: the 5 data pulses must all be at least 3 dB above the reference level of the existing signal. The rationale for this rule is that, since the new preamble occurs during a Mode S signal, it is virtually certain that the 5 data pulses will exist (namely as pulses of the existing signal). The revised requirement ensures that the pulses used for the new preamble detection are pulses generated by a stronger signal, and not the existing one.

I.4.1.3 Summary of Preamble Declaration Enhancements

The enhanced Extended Squitter waveform preamble detection technique described above provides for significant improvements in environments with very high levels of Mode A/C fruit. The improved algorithm should require only a modest increase in hardware complexity and/or software processing load and complexity compared to the existing algorithm

I.4.2 Enhanced Bit and Confidence Declaration

I.4.2.1 Overview

The current technique of declaring a bit based upon the higher of the two chips will generate bit errors in cases of higher level overlapping Mode A/C fruit (Figure I-2, part c). The use of amplitude to correlate the received pulse with the preamble pulse level will improve bit declaration accuracy. Four techniques have been investigated. One is a very simple approach that uses only the amplitude measured at the center of each chip. The remaining three use a more capable approach that takes advantage of all samples per chip that are taken to establish bit and confidence. Each of these techniques is described in the following paragraphs.

The following description of the center sample technique is intended to provide an example of processing performance needed to meet the requirements of Class A1 equipment. This description also serves as an introduction to the three more capable approaches. The center sample technique will not provide sufficient performance to meet requirements specified for Class A2 and A3 equipment in the test procedures of §2.4.4.4. The specified performance for these latter classes of equipment can only be meet by an approach that performs equivalently to one of the multi-sample techniques.

I.4.2.2 Use of Center Amplitude

An improvement in the declaration of Mode S data bits can be achieved if the actual amplitudes of the center samples of the '1' and '0' chips can be measured for each data position, rather than just a comparison of which chip sample is greater. All Mode S pulses, including those of the preamble, have approximately the same level (within 1 or 2 dB). Thus if the preamble level is measured, the expected level of each data pulse will be known. Then if both center samples of a data position are above threshold, but only one is within a ±3 dB band centered at the preamble level, it would be reasonable to assume that the corresponding chip is the correct Mode S pulse location. This is illustrated in Figure I-2, part d.

Figure I-3 illustrates the new data and confidence declaration algorithms that result when the actual sample amplitudes can be measured, and both samples are above threshold. As shown in Figure I-3, a bit is high confidence when 1 and only 1 of the two samples correlate with the preamble level. The correlating sample, rather than the larger sample, is declared to be the true data value. If both samples, or neither sample, correlates with the preamble, a low confidence bit is declared. In this case, the larger sample is selected as the bit value, as in the current technique.

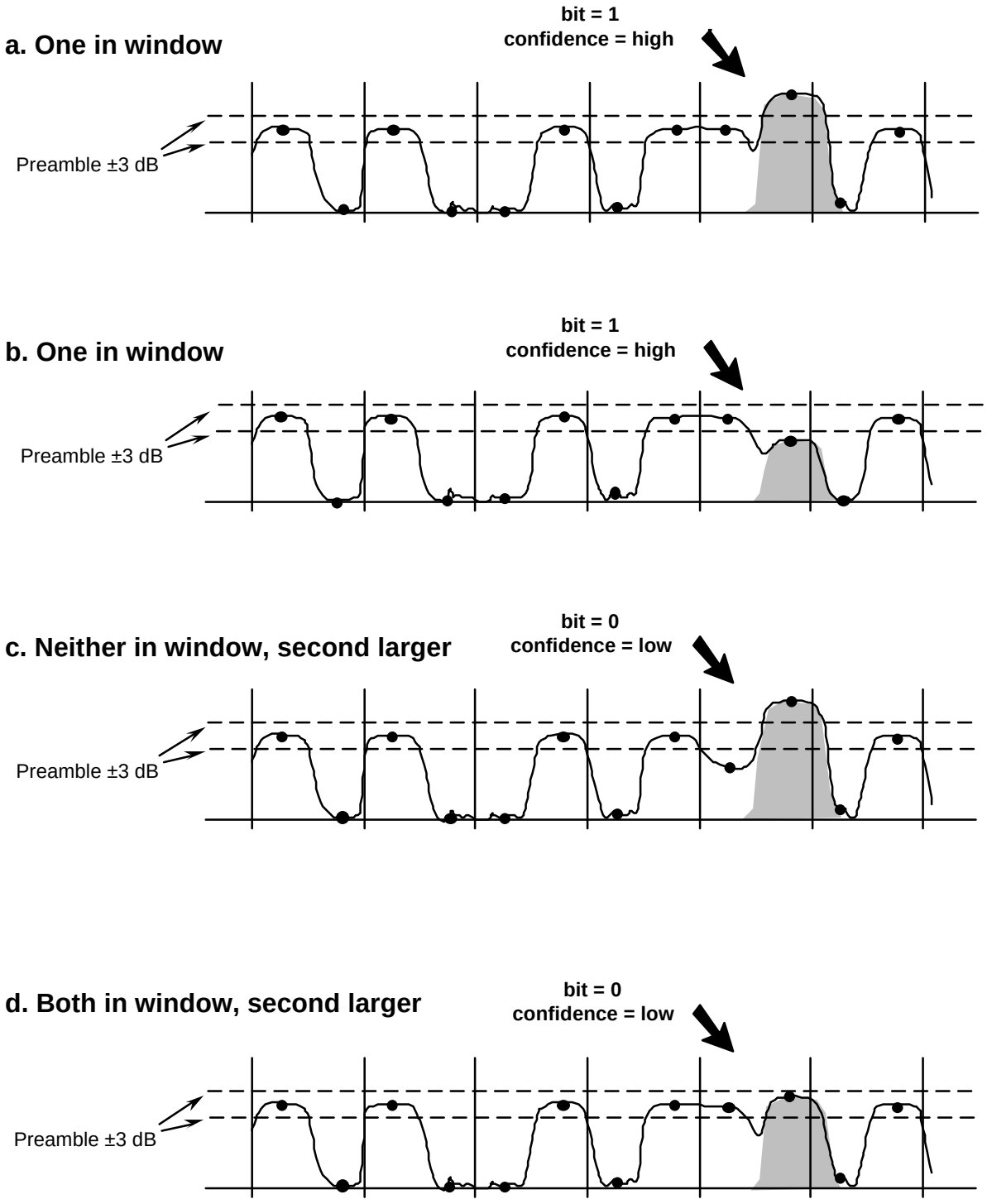

Figure I-3: Center Amplitude Bit and Confidence Declaration

The major advantage of this scheme is the significant reduction in low confidence and bit errors generated for the Mode S message. Presently, any data position with both samples above threshold is declared low confidence. This translates to a low confidence

declaration whenever an above-threshold Mode A/C fruit overlaps a Mode S empty chip position. If the Mode A/C signal is of greater power, the bit position is declared in error.

With the center amplitude technique, a low confidence bit is declared only when the Mode A/C fruit is within 3 dB of the Mode S signal. Mode A/C fruit of either lower or higher power in general cause neither errors nor low confidence bits. Thus the region of concern for Mode A/C aircraft is reduced from all the aircraft at closer range to just those at approximately co-range. This is particularly significant when the system is attempting to listen to far-range Mode S aircraft.

It is occasionally possible for this new algorithm to produce high confidence bit errors. The most likely case occurs when two Mode A/C pulses overlay the same Mode S bit position. If one overlaps the data pulse, and drives its sample out of the preamble window, then an error will result if the other Mode A/C pulse lands on the other chip and its amplitude is within the preamble window. Such occurrences will in general create a rejected message, rather than an undetected error.

I.4.2.3 Use of Multiple Amplitude Samples

The above amplitude data declaration approach can be improved if all 10 samples (5 per chip) that are taken for each Mode S bit position are utilized in the decision process. In particular, the event of both center samples within the preamble window can often be resolved. For example, consider the typical such situation depicted in Figure I-2, part e. Although both center samples are within the window, the fact that the earlier chip has all samples within the window, whereas the later chip has several samples below the window suggests that the signal is present in the earlier chip. This error situation can often be rectified when all 10 samples are examined. Also, in some cases when an interference pulse overlaps a signal pulse, the small frequency difference will produce variations in amplitude, whereas the signal by itself would be more constant, and such patterns can be useful in declaring the bit and confidence.

I.4.2.3.1 Baseline Multi-Sample Technique

The multi-sample enhanced bit and confidence declaration technique makes use of all 10 samples for each Mode S bit position to determine the bit and confidence values. Sample amplitudes in each chip are compared to the amplitude reference level established by the preamble to quantify the number of samples in each chip that:

(a) match the preamble amplitude indicating the presence of a pulse, or

(b) are significantly lower in amplitude indicating a lack of transmitted energy.

The first step is to establish an amplitude window that will include samples that are within +/- 3 dB of the preamble reference level and a minimum amplitude threshold set to 6 dB below the reference level. Samples that fall within the window are considered to match the preamble and samples that are below the minimum threshold are considered to indicate a lack of transmitted energy. The samples are categorized as follows:

A: within the +/- 3 dB preamble window

B: below threshold (6 dB or more below the preamble)

The second step is to count the number of samples in each chip that are of each category. Less weight is given to the samples near the transitional areas of each chip (the transitional samples are the first and last samples of each chip). To facilitate this, samples other than those at each end will count double. Therefore, with weighting factored in, and a 10 MHz sampling rate, the counts for each category will range from 0 to 8 for each chip (1 sample at each end + 3 samples in-between x 2). The four counts are summarized as follows:

1ChipTypeA = #of weighted samples in the 1 chip of type A (Match Preamble)

1ChipTypeB = #of weighted samples in the 1 chip of type B (Lack energy)

0ChipTypeA = #of weighted samples in the 0 chip of type A (Match Preamble)

0ChipTypeB = #of weighted samples in the 0 chip of type B (Lack Energy)

Next, two equations using the above counts will produce two scores that indicate how well the sample pattern matches a transmitted 0 and how well the sample pattern matches a transmitted 1. The equations are as follows:

1Score = 1ChipTypeA – 0ChipTypeA + 0ChipTypeB – 1ChipTypeB

0Score = 0ChipTypeA – 1ChipTypeA + 1ChipTypeB – 0ChipTypeB

The highest score determines the bit value. In the case of a tie, the bit defaults to zero (0). If the difference is 3 or more the bit is high confidence. The confidence threshold of 3 was determined by testing the algorithm with a 10 MHz sampling rate with thousands of iterations with a high fruit rate. If the algorithm is applied with a different sampling rate, the appropriate confidence threshold may need to be determined under similar test conditions.

I.4.2.3.2 Multi-Sample Technique With Reduced Table Lookup

To take advantage of the differences between a pure signal and a combination of signal plus interference, a technique was developed based on a lookup table, whose contents are derived from many runs of a simulation. Specifically, each of the 10 samples is quantized into four levels:

0: below threshold (-6 dB relative to the preamble)

1: above threshold but below the +/- 3 dB preamble window

2: within the +/- 3 dB preamble window

3: above the +/- 3 dB preamble window

Since there are 10 samples, with 4 possible values each, a Mode S data position can have 4^{10} = 1048576 (1M) different sample patterns. Two 1-bit tables, each stored in a 1M x 1 ROM, are defined over the set of patterns: the first declaring the bit position to be a '1' or '0', the second high or low confidence. Once the pattern existing for a given bit is determined, two table lookups supply the proper declaration. If higher sampling rates are used, the number of sample patterns and the table size will increase exponentially.

These tables are generated by running millions of simulations of Mode S messages in 40,000 fruit per second environments. For each bit of each trial, the pattern and the correct Mode S bit value are noted. The result could be, for example, 5876 examples of pattern 16453, of which 5477 occurred when the Mode S bit was a "1." The table values are then defined as follows, assuming an "uncertainty parameter" value of 10%:

H1: 90% or more of the samples occurred when the bit was a '1'

L1: 50% - 90% of the samples occurred when the bit was a '1'

L0: 10% - 50% of the samples occurred when the bit was a '1'

H0: 10% or fewer of the samples occurred when the bit was a '1'.

Since pulse shapes are critical to this method, live data verification of the table entries is required to establish these values.

I.4.2.3.3 Multi-Sample Technique Without Table Lookup

The above 10 sample approach requires lookup tables of size 1M, adding cost to the hardware implementation of the decoder. A variation of this method that only requires tables of size 1K, called the 5-5 approach, has been designed to reduce this expense.

The 5-5 method forms two estimates of the bit data and confidence values, one using the odd samples (1-3-5-7-9) and the other using the even samples (2-4-6-8-10); the final decision is then a combination of the individual estimates. Since each set includes samples in both chip positions, pattern matching is still possible, although the fineness of the pattern variation is cut in half. Since only 5 samples are in each set, and each sample is quantized to the same 4 levels as above, 4^5 = 1024 patterns are possible for each set.

To counteract the loss of resolution, and to aid in the combining operation, 3 levels of confidence (high, medium, and low) are defined for each pattern. Following the simulation generation scheme described above, the table values are defined as follows:

H1: 90% or more of the samples occurred when the bit was a '1'

M1: 70% - 90% of the samples occurred when the bit was a '1'

L1: 50% - 70% of the samples occurred when the bit was a '1'

L0: 30% - 50% of the samples occurred when the bit was a '1'

M0: 10% - 30% of the samples occurred when the bit was a '1'

H0: 10% or fewer of the samples occurred when the bit was a '1'.

The values of 10% and 30% are parameters; the value of 30% was selected to provide the performance discussed below. Since 3 confidence levels now exist, the confidence lookup table for each sample set is sized 1024x2. The total table requirement for data and confidence, for the two sample sets, is thus 1024x6.

Once the values and confidences are determined for each set of samples, odd and even, the composite values actually declared for the bit are found according to Table I-1.

Table I-1: Combining Odd and Even Outputs

Odd	Even					
	H1	M1	L1	H0	M0	L0
H1	H1	H1	H1	L0	H1	H1
M1	H1	H1	L1	H0	L0	L1
L1	H1	L1	L1	H0	L0	L0
H0	L0	H0	H0	H0	H0	H0
M0	H1	L0	L0	H0	H0	L0
L0	H1	L1	L0	H0	L0	L0

Note that if either sample set is high confidence, that set's data value rules unless the samples conflict and are both high confidence. Also notice that two agreeing medium confidence samples produce a high confidence result. With the parameter for medium confidence set at 30%, the probability of error for medium confidence agreement is 0.3 x 0.3 = 0.1, which matches the 10 sample probability of error for a high confidence decision. Finally, when the sample decisions conflict at the same confidence level, a low confidence 0 is declared for lack of a better estimate.

If a sampling rate other than 10 MHz is used, the number of samples from the odd and even samples and the resulting table sizes will be adjusted accordingly. For example, if a 8 MHz sampling rate is used, there are 4 samples in each set, and each sample is quantized to the same 4 levels as above, $4^4 = 256$ patterns are possible for each set. The total table requirement for data and confidence, for the two sample sets, is thus 256x6. If a 16 MHz sampling rate is used, there are 8 samples in each set, and each sample is quantized to the same 4 levels as above, $4^8 = 65{,}536$ (64K) patterns are possible for each set. The total table requirement for data and confidence, for the two sample sets, is thus 64Kx6. Once the values and confidences are determined for each set of samples, odd and even, the composite values are declared according to Table I-1 independent of the sampling rate.

I.4.3 Enhanced Error Detection and Correction Techniques

I.4.3.1 Overview

Three new error detection/correction techniques have been developed. The first, termed the "Conservative" technique is a variation on the current Sliding Window technique, intended to reduce the undetected error rate. The second, termed the "Whole Message" technique, models the effect of whole Mode A/C fruit on the bit and confidence declarations. The third, termed the "Brute Force" technique, performs a bounded

exhaustive search of all combinations of bit reversals for low confidence bits. The following sections describe these techniques in more detail.

I.4.3.2 Conservative Technique

The sliding window technique is suitable for the low fruit environments of a rotating beam antenna (long range, narrow beam) or a TCAS (omni-directional, short range). However, the fruit environment for the long-range air-to-air application (which uses omni-directional antennas) may be very severe and therefore, the sliding window technique cannot be used because of undetected error considerations.

For the very severe fruit environments, a simpler approach, known as the conservative technique is used. Using this technique, error correction is only attempted if all of the low confidence bits in the message are within a 24-bit window, and there are no more than 12 low confidence bits. This constraint limits the application of error correction to signals that nominally had only a single overlapping stronger Mode A/C fruit. This is a conservative approach in that the conditions for attempting error correction are much more restrictive than with sliding window. It produces a lower level of successful error correction since it does not attempt to correct messages with multiple Mode A/C overlaps. However, it produces a very low undetected error rate, as intended.

If the conditions for applying the conservative technique are met, the error syndrome is generated for the window position and (as for the sliding window technique) a check is made to see if the ones (1) in the error syndrome correspond to low confidence bits in the window. If so, error correction is accomplished. If not, the process is terminated.

If the low confidence bits span less than 24-bits, more than one window could be defined to span them. This will not effect the error correction action, since regardless of the 24-bit window selected to span the low confidence bits, the ones (1) in the error syndrome will identify the same message bits. That is, if the window is moved one bit, the error syndrome will shift by one bit.

Note that in the above description, there is only one possible successful error correction possibility. Regardless of the specific 24-bit window position, the same message bits are identified. All of the bits corresponding to a one (1) in the error syndrome must be complemented. This can only happen if they are all low confidence. Therefore, there is at most, one correctable error pattern that can be achieved with the conservative error correction technique.

I.4.3.3 Whole Message Error Detection and Correction Technique

The limitation of the current error correction technique described in I.3.3 of being able to handle only a single overlapping Mode A/C fruit has led to the development of the Whole Message error correction technique, which can be applied when the Conservative technique has failed. This technique is designed to handle up to five overlapping fruit, provided they do not overlap each other, hence creating "individual interference regions." This technique is applicable only when the center sampling technique is used (§I.4.2.2).

As described in I.3.2, the correctness of low confidence bits depend upon the amplitude of the Mode A/C interference relative to the Mode S squitter: if the Mode A/C fruit is stronger than the Mode S signal, any low confidence bit will usually be declared in error, while if the Mode S signal is stronger, any low confidence bit will usually be declared correctly. This observation leads to the following hypothesis, which serves as the basis of the Whole Message error correction routine:

> *For a given Mode A/C interference region, either all low confidence bits will be correct, or all low confidence bits will be wrong.*

Although this hypothesis is generally correct, it should be noted that three effects can lead to its violation:

1. If the interfering Mode A/C fruit is at approximately the same power level as the Mode S signal, or

2. If the interfering Mode A/C fruit pulses are wider than 0.50 microseconds (the spec allows up to 0.55 microseconds, and some out-of-spec transponders even go beyond this value), or

3. If the Mode S signal is near enough to the noise level for noise to corrupt some of the samples.

The Whole Message algorithm attempts to divide the Mode S message into disjoint 24-bit regions, one for each presumed Mode A/C interfering fruit. Then the syndrome of each region, assuming all bits to be in error, is generated (see below in §I.4.3.4 for an explanation of syndrome generation for bit sets). Finally, all possible combinations of syndromes are considered. If one and only one combination matches the Mode S syndrome, all low confidence bits within the interference regions corresponding to the winning combination are reversed. Flowcharts for the Whole Message error correction technique are presented in Figure I-4, parts a and b. Figure I-5 presents an example of the application of this method.

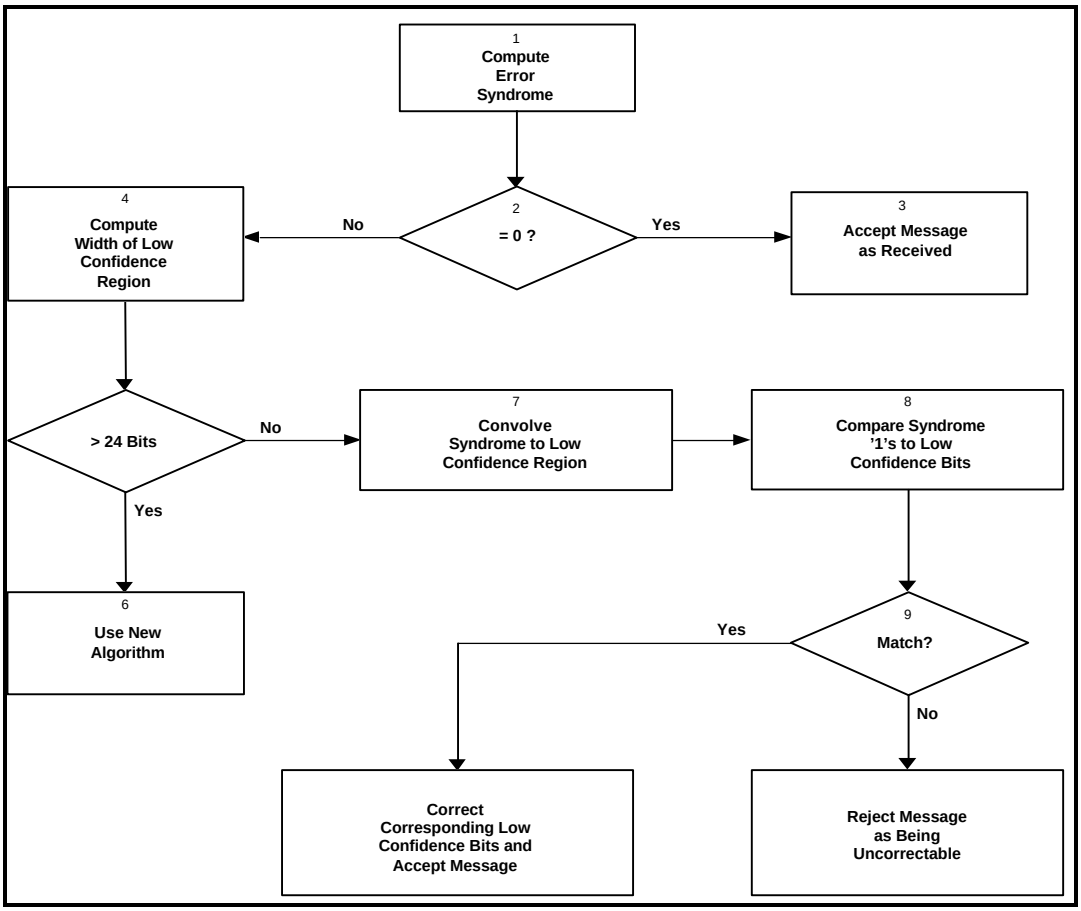

Figure I-4a: Whole Message Top Level Error Correction Algorithm

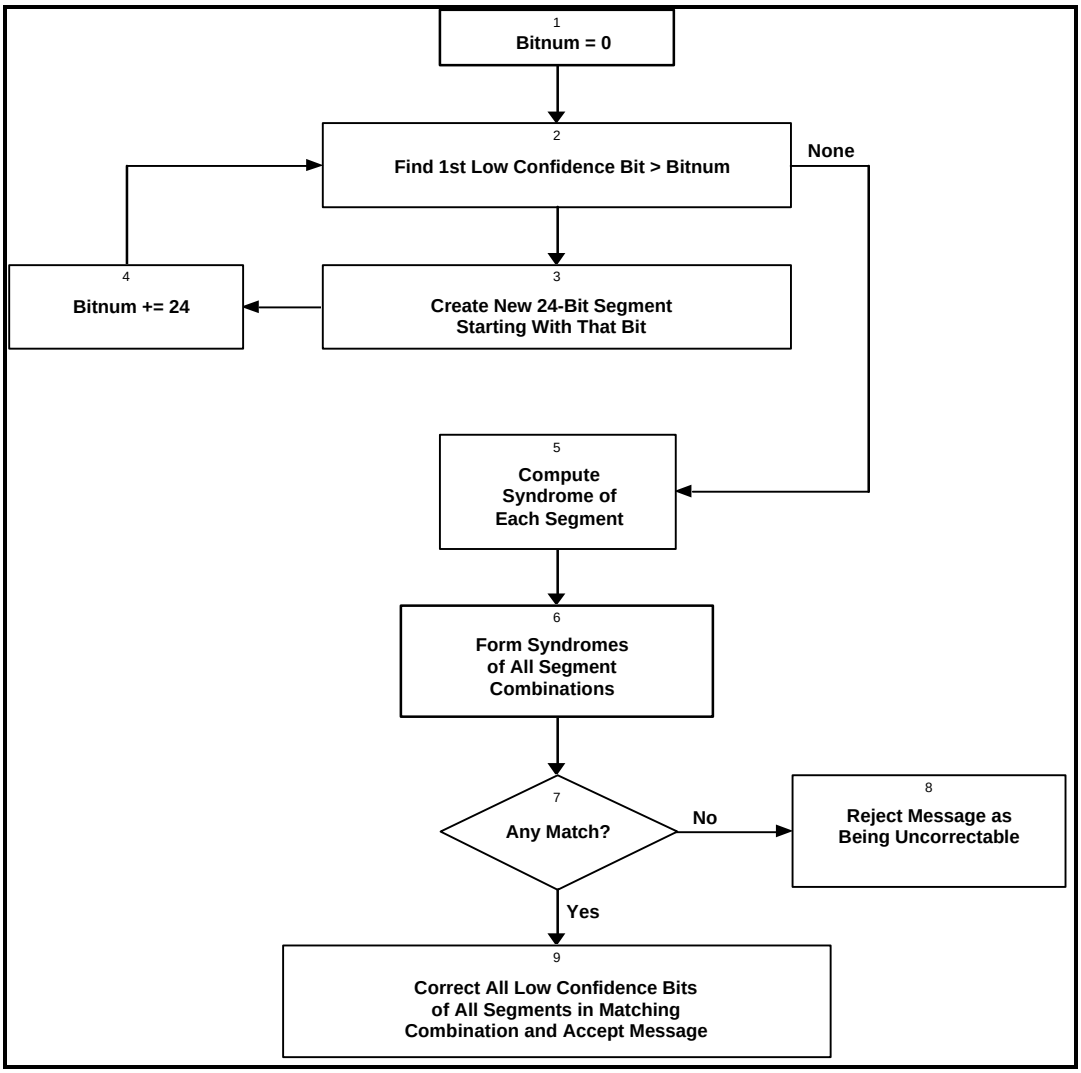

Figure I-4b: Whole Message Detailed Error Correction Algorithm

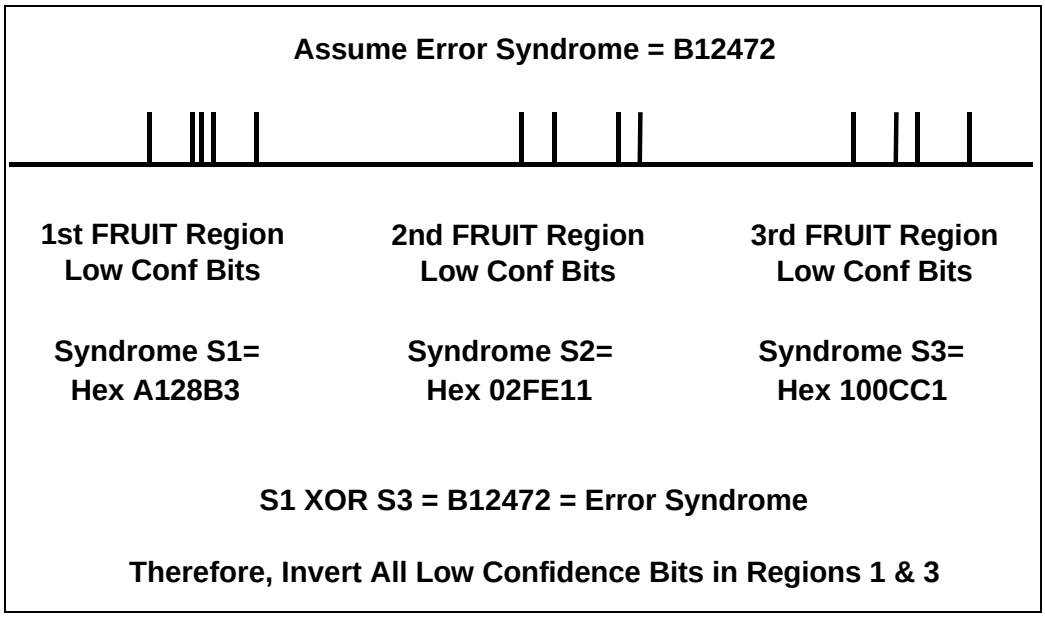

Figure I-5: Whole Message Error Correction Example

I.4.3.4 Brute Force Error Correction Technique

If the bit declaration algorithm has performed its function properly, all errors in Mode S data values will reside in bits declared low confidence. If this is true, a simple approach to error correction is to try all possible combination of low confidence bits, and accept the set that matches the error syndrome (provided only one success is discovered). For obvious reasons, this method has been named the Brute Force Technique. It is applicable to any method of data and confidence declaration, with or without amplitude. As illustrated in Figure I-6, Brute Force error correction is applied after the other techniques have failed.

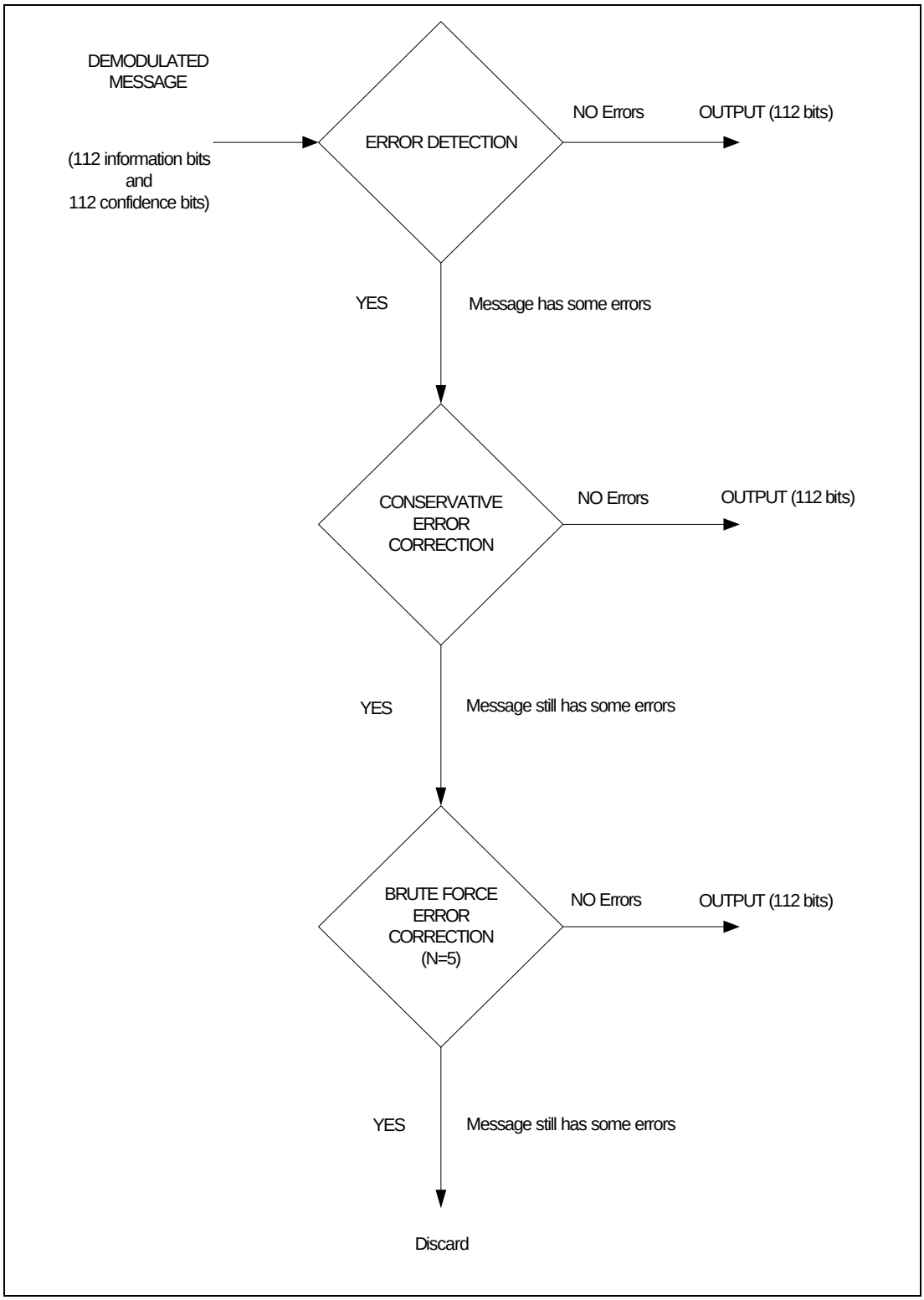

Figure I-6: Error Detection and Correction

Implementation of this technique depends upon the fact that each Mode S bit position corresponds to a unique syndrome, and that sets of bits produce a syndrome that is the exclusive OR of all individual bit syndromes. For example, if bit 1 is the only bit declared in error, the error syndrome at the receiver will be hex 3935EA, while bit 31 produces hex FDB444, and bit 111 has syndrome hex 000002. Thus if those three bits are all declared in error, the error syndrome will be calculated to be hex C481AC. The table of individual bit syndromes is pre-calculated and stored in the receiver.

It is possible for two or more subsets of the low confidence bits to match the syndrome. In such cases, the message is rejected, and no harm is done. However, if a high confidence bit has been declared in error, and a single subset of the low confidence bits matches the syndrome, the message will be "corrected" to the wrong message, producing an undetected error. (If no subset matches the syndrome, it must be true that a high confidence bit error has been made, and the message is rejected.)

Clearly, for processing time and error bounding reasons, the maximum number of low confidence bits to process must be limited. The number of cases to consider is given by 2^{n}. If **n** low confidence bits exist for a message; this grows exponentially with **n** (32 at **n** = 5, 4096 at **n** = 12). The undetected error rate is proportional to the number of cases, and thus also grows exponentially with **n**. Fortunately, the Hamming distance of 6 for the Mode S parity code implies that undetected errors are essentially zero if **n**</=5 is enforced. For this reason, a value of **n** = 5 has been used in the development of the brute force algorithm.

I.5 Improved Reception Performance in a High Fruit Environment

For improved reception performance in a high fruit environment, the optimum configuration has been found to be:

1. Enhanced preamble detection (§I.4.1)
2. Bit and confidence declaration based on the Baseline Multi-sample Technique (§I.4.2.3.1)
3. Error detection using the Mode S 24-bit CRC technique (Ref. RTCA DO-185A)
4. First pass error correction using the conservative technique (§I.4.3.2)
5. Second pass error correction using the brute force technique with n=5 (§I.4.3.4)

The process proceeds as follows. Preamble detection and bit and confidence declaration are performed. Next, Mode S error detection is applied. If the message passes, the process ends, and the message is delivered. If an error is detected, then conservative error correction is applied, and if a correction results, then the process ends, and the message is delivered. If the constraint for conservative correction is not satisfied, then the brute force technique is applied.

The above configuration was used as the basis for the performance required for enhanced squitter reception as specified in §2.2.4.4.

I.6 Summary

New techniques have been developed for enhancing reception of Extended Squitters in environments of high interference. The new techniques include improvements in preamble detection, improvements in declaration of information bits and confidence bits within the squitter message, and improvements in error detection/correction.

These developments were originally carried out using pulse-level simulation to assess the resulting performance. Subsequently flight tests have been conducted using these techniques, making comparisons relative to the current techniques. Both the simulation and flight test results indicate that substantial improvements in performance are achievable using these techniques when operating in an environment of high interference.

APPENDIX I REFERENCES

I-1. J. L. Gertz, "Fundamentals of Mode S Parity Coding," M I T Lincoln Laboratory Project Report ATC-117, April 1984.

This page intentionally left blank.